خوارزمية البحث في العمق Depth First Search DFS

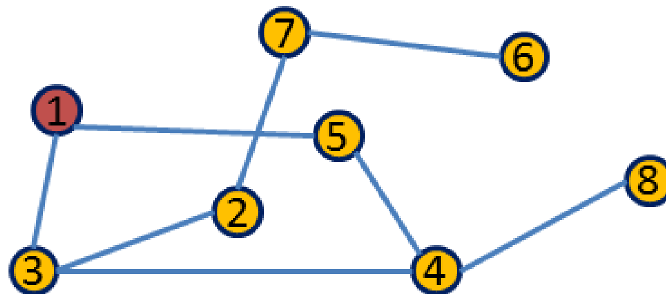
ملاحظة: يعتمد البحث في DFS على مبدأ المكدس Stack

ملاحظة: يؤمن الاستدعاء العودي في كود الخوارزمية مبدأ المكدس ولا الحاجة للتصريح عنه في بنية معطيات جديدة.

مبدأ العمل:

- سننشئ مصفوفة من نوع linkedlist وكل عنصر فيها هو linkedlist.
- نستطيع معرفة هل عقدة ما مزارة أم لا بواسطة مصفوفة اسمها visited من نوع boolean وجميع العقد في البداية غير مزارة.
- عقدة البداية s بالتأكيد غير مزارة ، لذا سنحولها إلى مزارة ونطبعها
- طالما أن عقدة الجوار موجودة نفحصها هل هي غير مزارة؟ اذا كانت غير مزارة سنحولها إلى مزارة ثم ونطبعها.

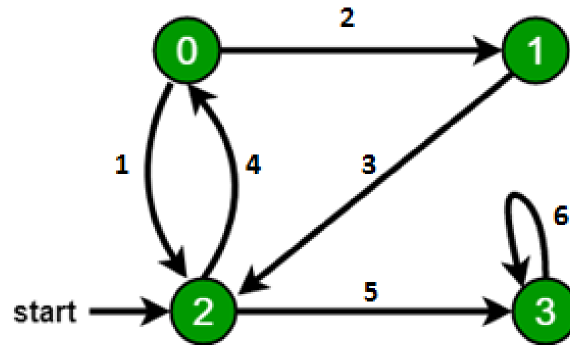
مثال: أوجد العبور (traverse) لهذا البيان بطريقة DFS بدءً بالعقدة 1.



الحل: بعد كتابة مصفوفة visited سيكون التجوال هو:

1 3 2 7 6 4 5 8

مثال: اكتب كود خوارزمية البحث بالعمق DFS بلغة C++، ثم أوجد التجوال للبيان التالي بدءً من العقدة 2، مع ملاحظة أن الأرقام الموجودة على الروابط هي لتوضيح تسلسل انشائها.



الحل:

```
#include <iostream>
#include <list>
using namespace std;

class Graph {
  int numVertices; // No. of vertices
  list<int> *adjLists;
  bool *visited;

public:
  Graph(int numVertices);
  void addEdge(int src, int dest);
  void DFS(int s);
};

// Initialize graph
Graph::Graph(int vertices) {
  numVertices = vertices;
  adjLists = new list<int>[vertices];
  visited = new bool[vertices];
  for(int i = 0; i < vertices; i++)
           visited[i] = false;
}

// Add edges
void Graph::addEdge(int src, int dest) {
  adjLists[src].push_back(dest);
}

// DFS algorithm
void Graph::DFS(int s) {
  visited[s] = true;
  //list<int> adjList = adjLists[s];
  cout << s << " ";

  list<int>::iterator i;
  for (i = adjLists[s].begin(); i != adjLists[s].end(); i++)
```

```
    if (!visited[*i])
      DFS(*i);
}

int main()
{
     // Create a graph given in the above diagram
     Graph g(4);
     g.addEdge(0, 2);
     g.addEdge(0, 1);
     g.addEdge(1, 2);
     g.addEdge(2, 0);
     g.addEdge(2, 3);
     g.addEdge(3, 3);

     cout << "Following is Depth First Traversal"
                " (starting from vertex 2) \n";
     g.DFS(2);
     system("pause");
     return 0;
}
```

التجوال بدءاً من العقدة 2 هو:

2 0 1 3